

Department of Computer Science and Engineering
The University of Texas at Arlington

Aegle

Outreach Storage System

Team Members:

Joseph Finnegan

Karla Hernandez

Joe Martinez

Carlos Torres

Document Revision History

Revision Number	Revision Date	Description	Rationale
0.1	10/08/2014	SRS First Draft	Initial draft for OSS system
0.2	10/28/2014	SRS First Draft Revisions	Implement instructor's and TA's feedback
0.3	11/04/2014	SRS Peer Review Revisions	Implemented peer review feedback
1.0	11/05/2014	SRS Gate Review Draft	Final changes prior to gate review.
2.0	11/16/2015	SRS Baseline	Final changes after gate review.

Table of Contents

1. Product Concept.....	8
1.1 Purpose and Use	8
1.2 Intended Audience.....	9
2. Product Description and Functional Overview	10
2.1 Features and Functions.....	10
2.2 External Inputs and Outputs	11
2.3 Product Interfaces.....	13
2.4 Definitions, Users, and Assumptions.....	21
3. Customer Requirements.....	24
3.1 Keep track Items and Crates by the System	24
3.2 System Description of Items	24
3.3 System Description of Crates.....	25
3.4 Search Function for Items, Crates and Projects.	25
3.5 Locating Item Inside a Crate	25
3.6 Item Management by the Administrators.....	26
3.7 Crate Management by the Administrators	26
3.8 Project Management by Administrators.....	26
3.9 System Interaction by Administrators.....	27
3.10 System Interaction by Registered Users	27
3.11 Registration/Login System	28
3.12 Web-Based Accessible Application	28
3.13 Locating Crate Inside the Storage Room.....	28
4. Packaging Requirements.....	29
4.1 Included Hardware Components.....	29
4.2 Installation Manual.....	29
4.3 Range of the RFID Reader Integrated Antenna	30
4.4 Software Components	30
4.5 RFID Reader Installation	30

4.6	Software Installation	31
5.	Performance Requirements	32
5.1	Check-in/Check-out Latency.....	32
5.2	Web Interface Response Latency	32
6.	Safety Requirements	33
6.1	Electrical Hazard	33
6.2	Signal Interference	33
7.	Maintenance and Support Requirements	34
7.1	User Manual	34
7.2	Troubleshooting Guide.....	34
7.3	Source Code Availability & Documentation	35
7.4	Database Interchangeability	35
8.	Other Requirements	36
8.1	Information Security	36
8.2	Connectivity Tolerance	36
9.	Acceptance Requirements.....	37
9.1	Verify User Account Registration.....	37
9.2	Verify Location of an Item Functionality	37
9.3	Verify Location of a Crate Functionality	38
9.4	Verify Search Functionality	38
9.5	Verify Item Management Functionality	39
9.6	Verify Crate Management Functionality.....	39
9.7	Verify Project Management Functionality	40
10.	Use Cases	41
10.1	Registering into the System.....	41
10.2	Logging into the System.....	41
10.3	Adding Item, Crate, or Project to the System	42
10.4	Editing Item, Crate, or Project in the System	42
10.5	Deleting Item, Crate, or Project from the System	43
10.6	Searching for Items and Crates.....	43
10.7	User Registration Confirmation by Administrator	44

10.8	Item/Crate Check-out Approval by Administrator	44
10.9	Item/Crate Checkout Request by Registered User	44
10.10	Item/Crate Checkout by Administrator.....	45
10.11	Item/Crate Check-in by Administrators and Registered Users.....	45
11.	Feasibility Assessment.....	49
11.1	Scope Analysis	49
11.2	Research.....	49
11.3	Technical Analysis	50
11.4	Cost Analysis	51
11.5	Resource Analysis	52
11.6	Schedule Analysis.....	53
12.	Future Items	58
12.1	Automatic Item Location Per Crate Inside the Storage Room	58
12.2	Automatic Crate Location Inside the Storage Room	58
12.3	Mobile Application	58

List of Figures

FIGURE	PAGE
Figure 1-1 Conceptual Diagram.....	9
Figure 2-1 Mockup prototype for Home page	13
Figure 2-2 Mockup prototype for Login / Register page	14
Figure 2-3 Mockup prototype for Search page	15
Figure 2-4 Mockup prototype for User Management page	16
Figure 2-5 Mockup prototype for Store Management page.....	17
Figure 2-6 Mockup prototype for Search Results page	18
Figure 2-7 Mockup prototype for Item Requests Management page	19
Figure 2-8 Mockup prototype for Item Request Form page	20
Figure 10-1 Login and Registration System	46
Figure 10-2 Inventory Check In / Out system	47
Figure 10-3 Inventory Management	48

List of Tables

TABLE	PAGE
Table 2-1 Table of inputs and outputs	12
Table 11-1 Preliminary Cost Analysis	51
Table 11-2 Function Point Table Breakdown	53
Table 11-3 Influence Multipliers	54
Table 11-4 Jones First Order Estimation	55
Table 11-5 Simplified Hybrid Approach	56

1. Product Concept

The following section describes the purpose, use, and intended user audience for the Outreach Storage System (OSS). The purpose of OSS is to manage project inventory for Dr. Tiernan the Outreach Coordinator at UTA. As Outreach Coordinator, Dr. Tiernan is responsible for organizing events and other activities to promote the College of Engineering to college bound students.

Many of these activities require different types of materials and tools to complete. When not in use, these materials are stored in a large room in Nedderman Hall. This storeroom until now has been organized manually, over time this had led to it becoming slowly disorganized as those with access to the storeroom have added and removed items haphazardly and without updating inventory counts. Because of this there is no precise accounting for what items and exactly how many of those items are in the storeroom at any one time.

This system will save time and money by keeping track of inventory and allowing for more efficient use of resources.

1.1 Purpose and Use

Managing product inventory is a problem that has been with human societies since the merchants of Sumeria. In the past this required meticulous record keeping and regular auditing to ensure accurate inventory records. OSS seeks to automate these processes and add new tasks specific to our customer's needs.

This system will use RFID technology to automate previously manual tasks. Inventory records will be updated as items are added or removed from storage crates, and their location within the storeroom will be recorded. OSS will also organize inventory into user defined projects, such as constructing a self-supporting tower made of dry spaghetti noodles, or categories, such as tools or crafts.

The goal of OSS will be to save the user the time and effort needed to find an item within a storeroom and to more effectively manage resources. By keeping accurate inventory counts, existing product can be used more efficiently and waste can be reduced.

1.2 Intended Audience

The target audience for this system is Dr. Tiernan, the Outreach Coordinator at UTA, and her assistants. The following conceptual diagram illustrates in a high level overview how the OSS will operate.

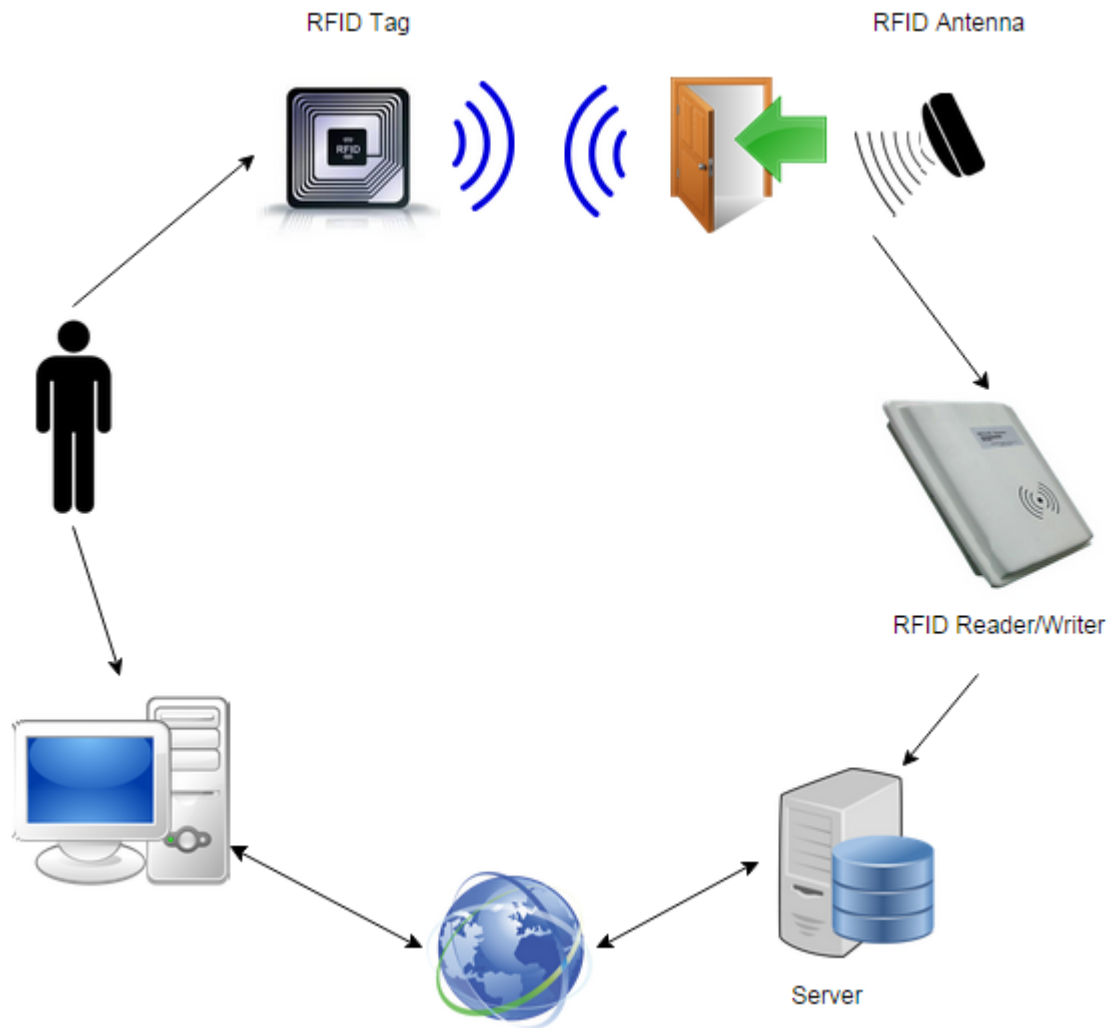


Figure 1-1 Conceptual Diagram

2. Product Description and Functional Overview

The following section provides the reader with an overview of OSS. The primary operational aspects of the product, from the perspectives of the end users, maintainers, and administrators, are defined here. The key features and functions found in OSS as well as critical user interactions and user interfaces are described in detail.

2.1 Features and Functions

OSS will consist of a server, RFID tags, and an RFID reader with integrated antenna. Every item that is to be automatically tracked by the system will have an RFID tag attached to it. The RFID reader will read these tags as items are added and removed from the storeroom. These readings will be sent to an inventory management system residing on a server, thereby keeping an accurate record of the total inventory.

Each item can optionally be assigned to a category that describes its function, such as electronics, crafts, etc. Each item will also optionally be assigned to a crate which will store the items; these crates will be arranged on shelves inside the storeroom. Crates can themselves also be assigned to either a category or a project. Finally, items may also be assigned to user defined projects of which they are a component of.

The system will implement an account based system for keeping track of individuals who have access and interact with the storeroom inventory. The system will have a single administrator who has access to all system functions and a separate class of users who are only allowed a limited number of actions. The administrator is the only account allowed to add new items, crates, and projects to the storeroom; they may also edit and delete items, crates, and projects from the system inventory. A user must submit an item request form if they wish to check out an item or crate from the storeroom, the administrator is then given the option to approve or deny the request.

2.2 External Inputs and Outputs

Name	Type	Description	Use
Register Account	Input	The user will input information to create an account	Creates a user account
Login	Input	Takes an email and password and matches them to an account already in existence	Logs a user into their account
Dashboard Page	Input / Output	Displays the most common system commands necessary to operate OSS	Main user interface
Manage Item / Crate	Input / Output	<ul style="list-style-type: none"> - Displays a form necessary for adding an object to the system inventory - Displays object data in an editable form - Presents user with a confirmation box to verify object deletion 	<ul style="list-style-type: none"> - Adds an item / crate to the inventory system - Item / crate will have their data updated - Item / crate will be removed from the system
Manage Project	Input / Output	<ul style="list-style-type: none"> - Displays a form necessary for adding a project to the system inventory - Displays project data in an editable form - Presents user with a confirmation box to verify project deletion 	<ul style="list-style-type: none"> - A project will be added to the system - A project will have their data updated - Project will be removed from the system
Item Description	Output	The system will show an item's data fields	User will see item information
Crate Description	Output	The system will show the crate's data fields	User will see crate information.
Project Description	Output	The system will show the project's data fields	User will see project description and the items necessary to perform it
Check Out Item / Crate	Input / Output	Displays a check out request form	The user will check out items / crates.
Check In Item / Crate	Input / Output	The system will check items/crates back in to the system	The user will check in items / crates.

Item Request	Input / Output	A user will enter information about the item they are requesting to check out	Sends an item request form to the Admin
Request Management	Input / Output	Displays a user's item request form, the admin will then approve/deny an item request	The item request will either be approved or denied.
Registration Management	Input / Output	Displays new user registrations that are currently pending	A new user's registration will either be approved or denied.
Search	Input / Output	User will input information for the item to be searched	Web page will display search results.
RFID Reader (antenna integrated)	Input	Reader will detect RFID tag information.	Tag information will be sent to the server
Server	Input / Output	Will store software system and connect to the host computer via the internet	Will store / retrieve all inventory information and system software
Database	Input / Output	The database stores all OSS related data and can be accessed or changed by the system.	The system shall be able to modify the database through queries.

Table 2-1 Table of inputs and outputs

2.3 Product Interfaces

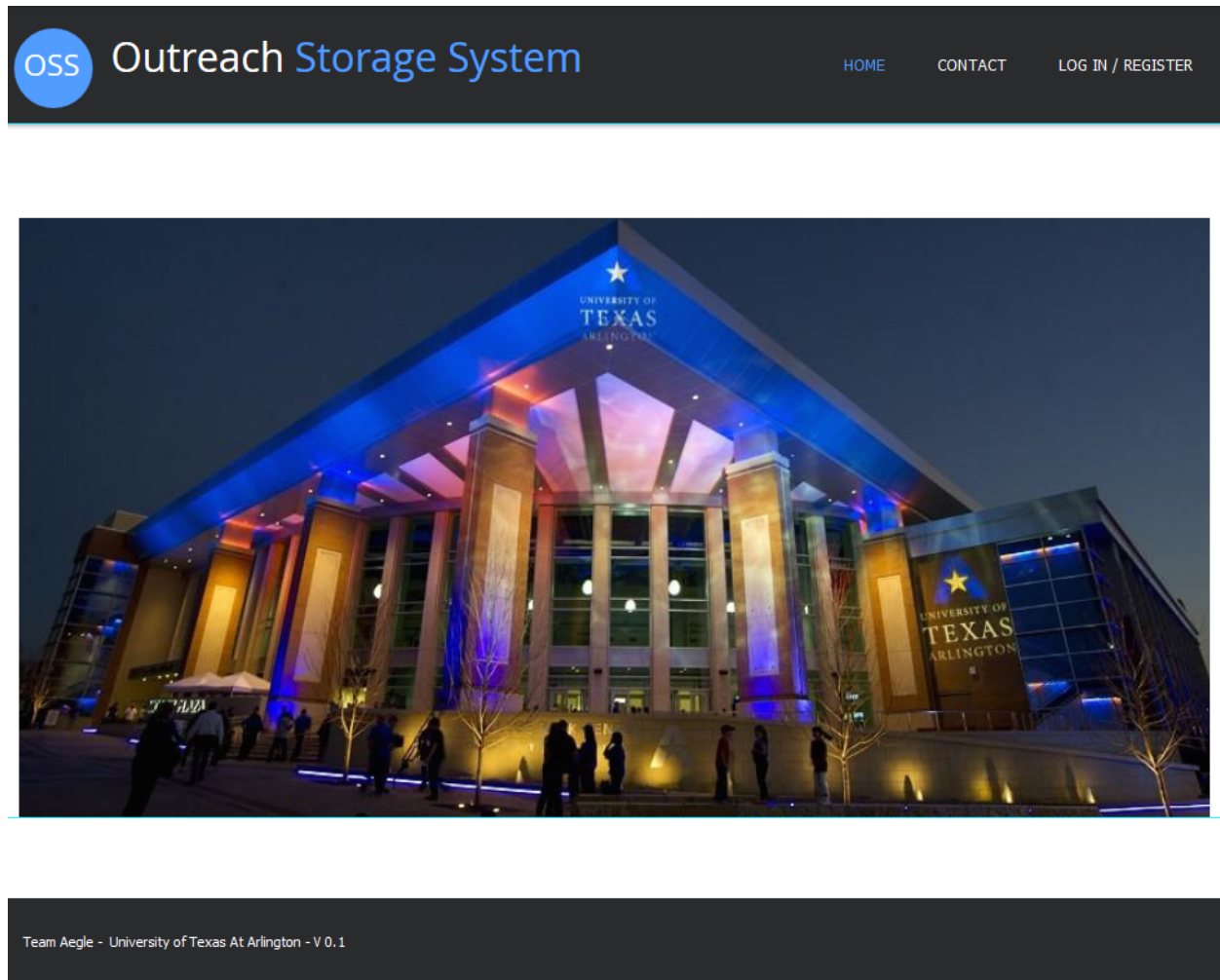
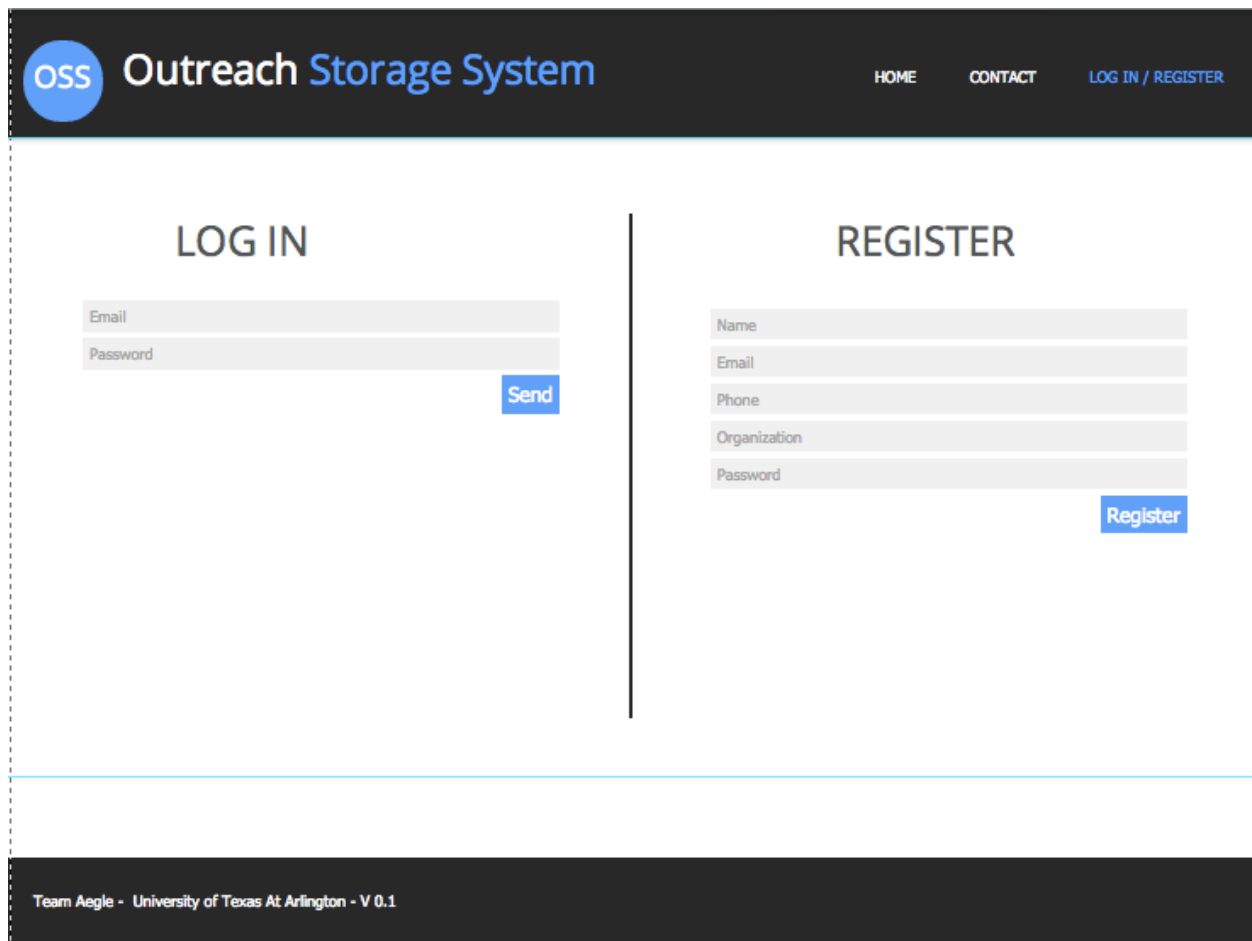


Figure 2-1 Mockup prototype for Home page



The mockup shows a web page for the Outreach Storage System (OSS). The header is dark blue with the OSS logo and navigation links: HOME, CONTACT, and LOG IN / REGISTER. The main content area is split into two columns by a vertical line. The left column is titled 'LOG IN' and contains two input fields for 'Email' and 'Password', followed by a blue 'Send' button. The right column is titled 'REGISTER' and contains four input fields for 'Name', 'Email', 'Phone', and 'Organization', followed by a blue 'Register' button. The footer is dark blue and contains the text 'Team Aegle - University of Texas At Arlington - V 0.1'.

oss Outreach Storage System

HOME CONTACT LOG IN / REGISTER

LOG IN

Email

Password

Send

REGISTER

Name

Email

Phone

Organization

Password

Register

Team Aegle - University of Texas At Arlington - V 0.1

Figure 2-2 Mockup prototype for Login / Register page

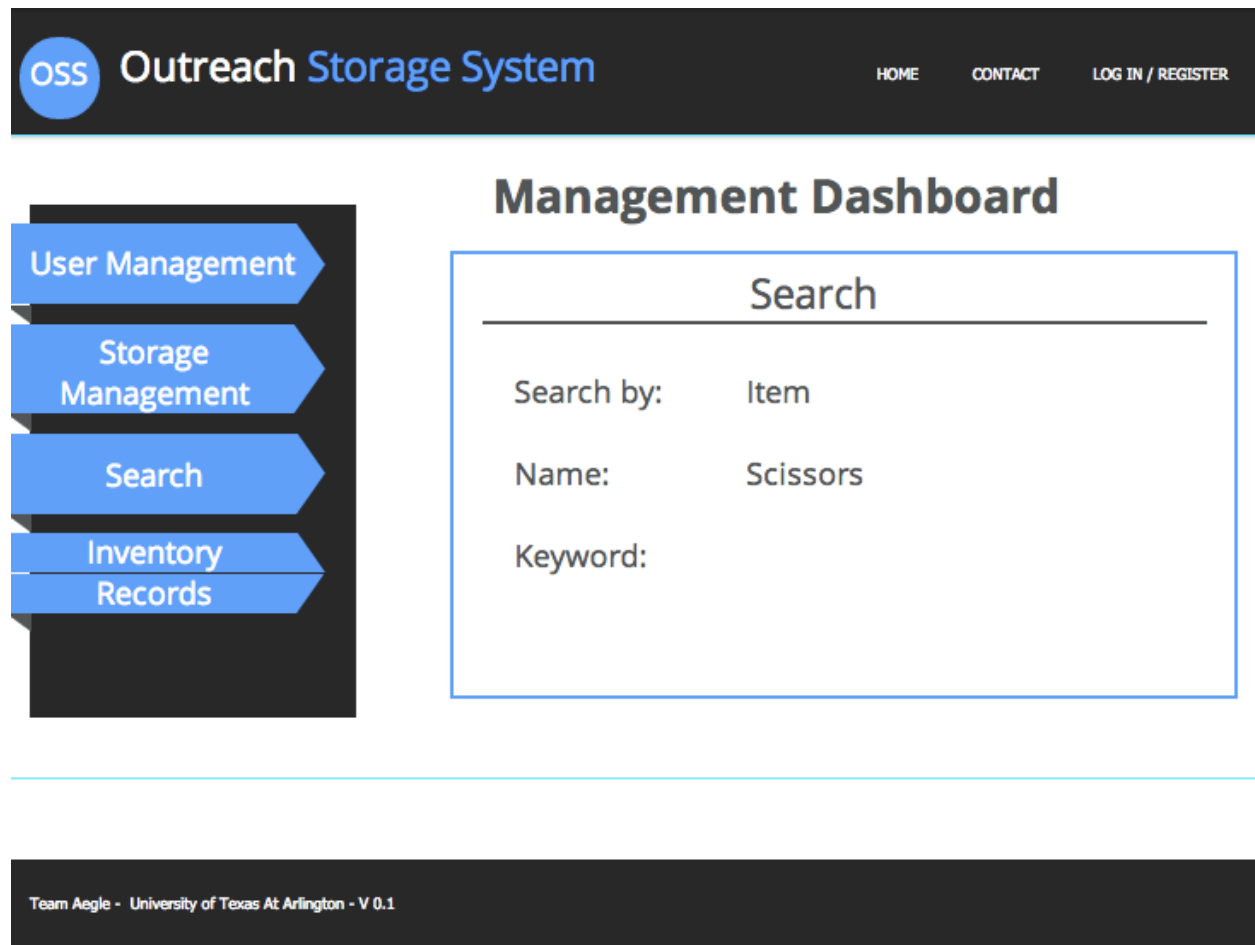


Figure 2-3 Mockup prototype for Search page

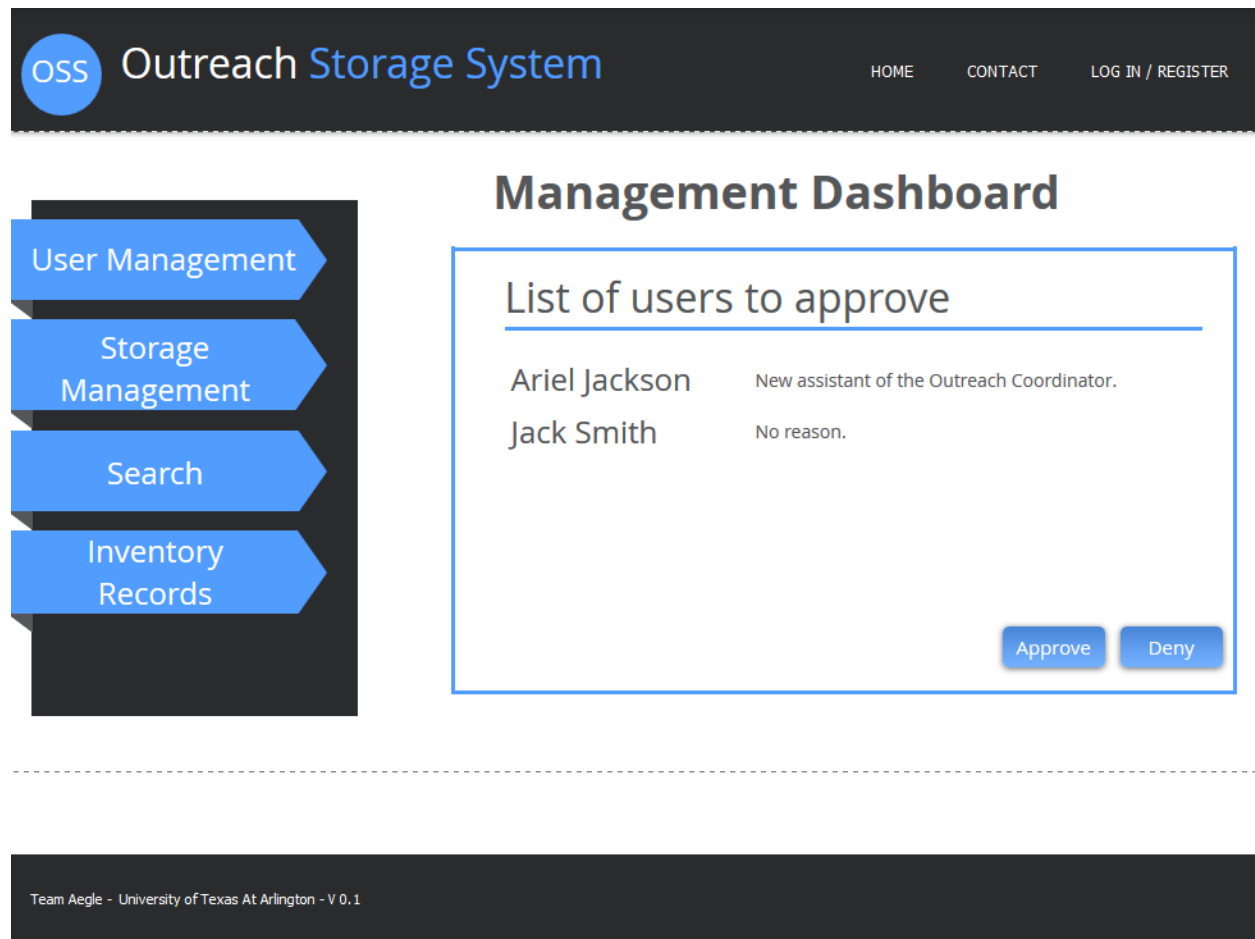


Figure 2-4 Mockup prototype for User Management page

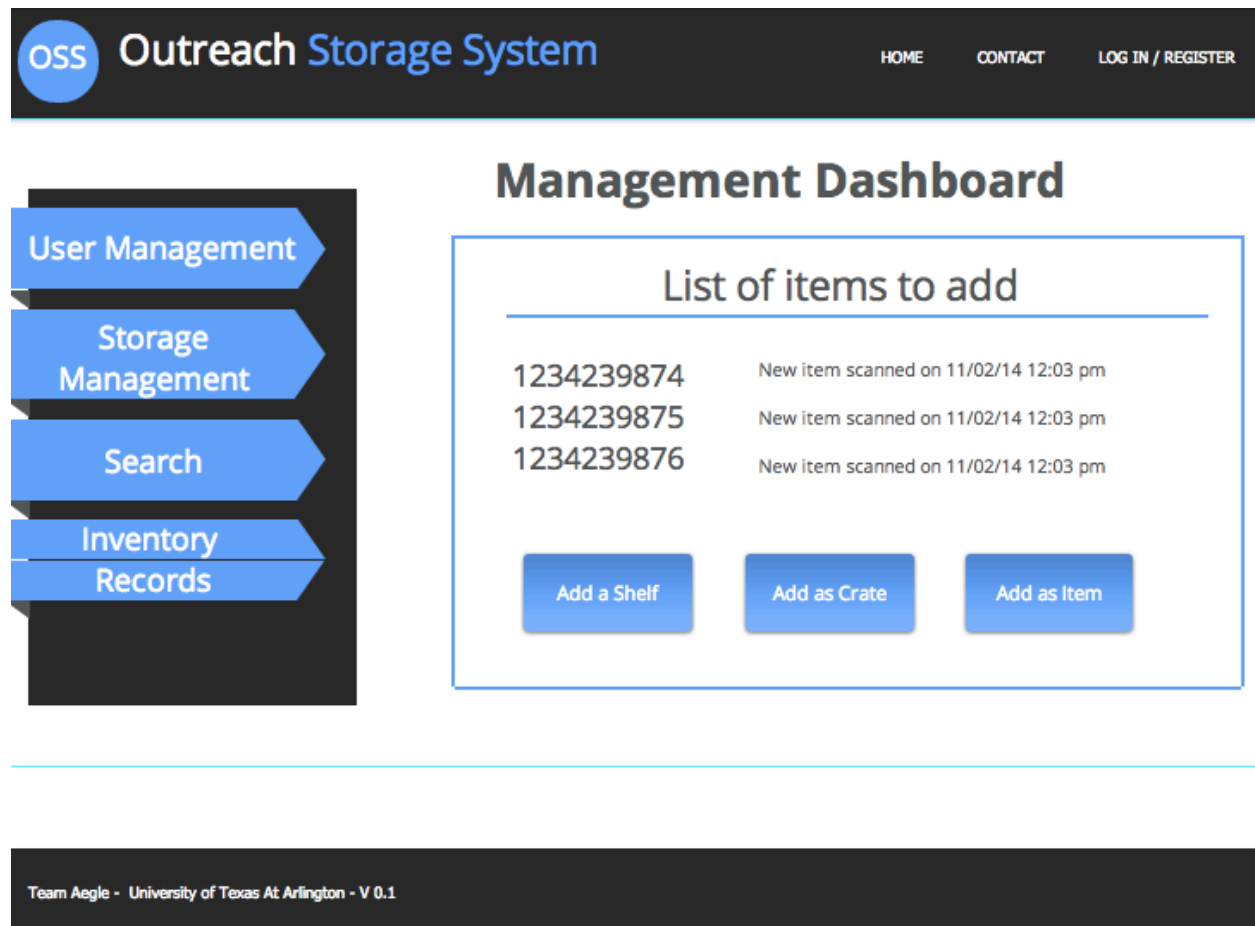


Figure 2-5 Mockup prototype for Store Management page

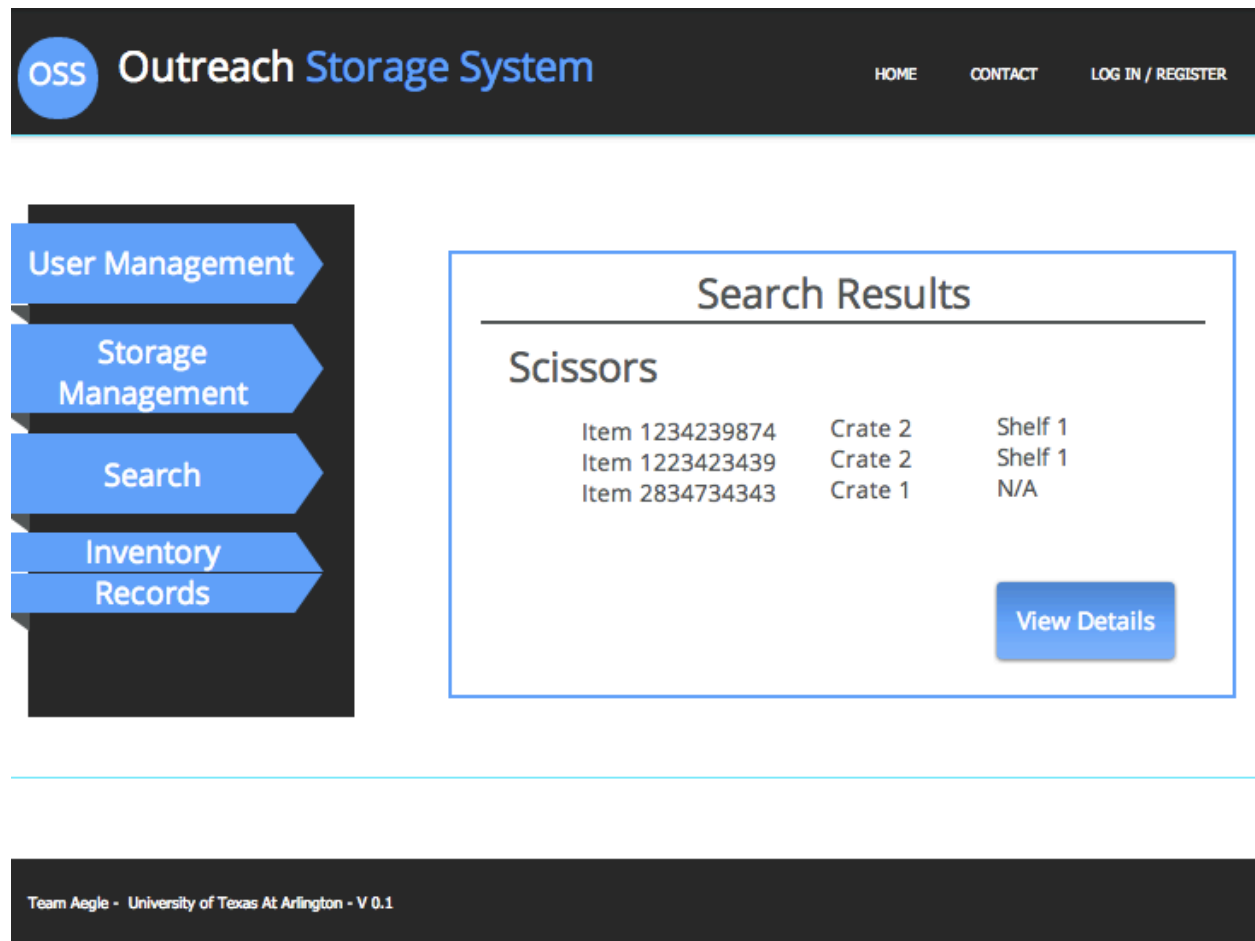


Figure 2-6 Mockup prototype for Search Results page

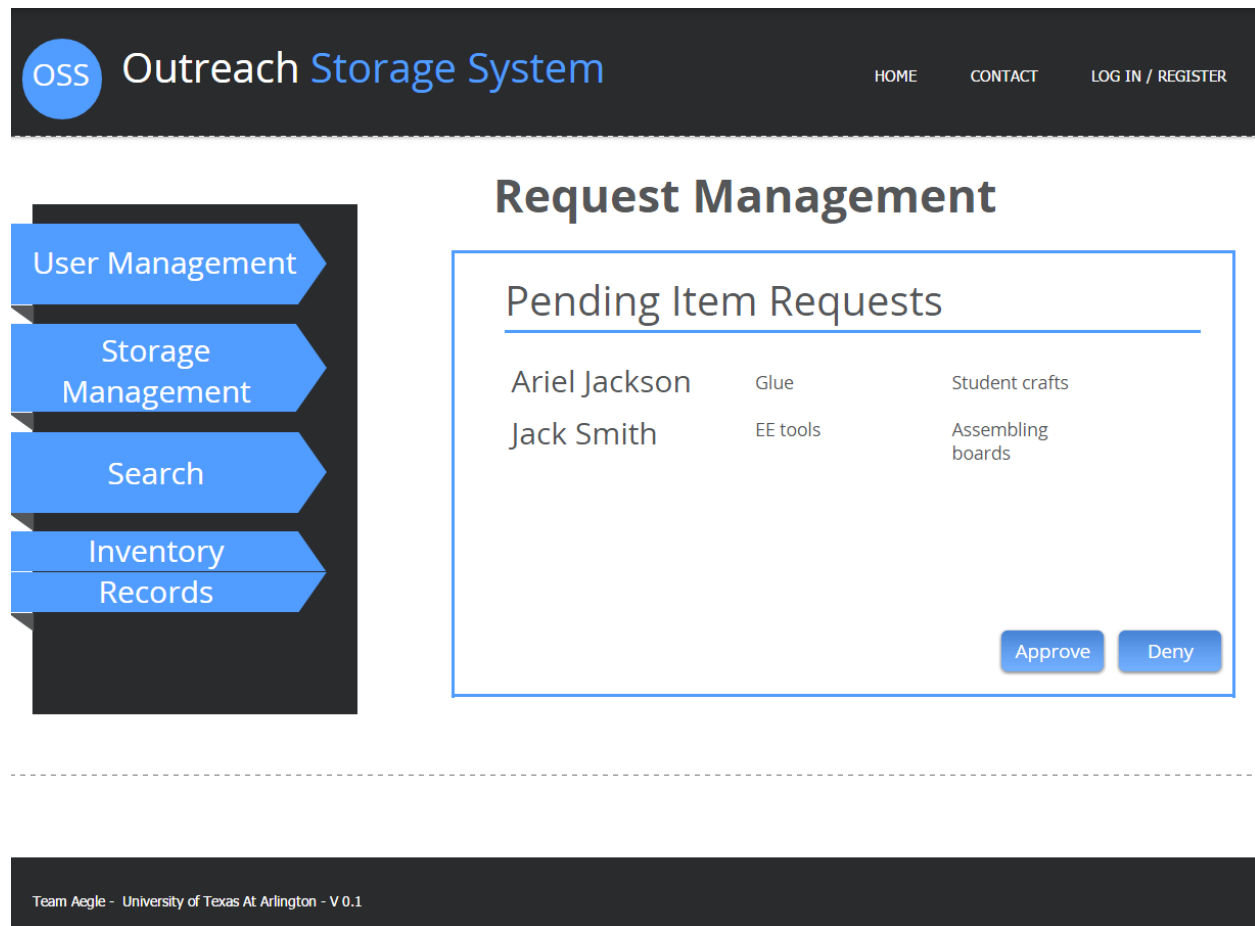


Figure 2-7 Mockup prototype for Item Requests Management page

oss Outreach Storage System

HOME CONTACT LOG IN / REGISTER

User Management

Storage Management

Search

Inventory Records

Item Request

Box of Sand Paper

Administrator

Registered User *

Requestor

Checkout Date

Date of Return

Reason

Send

Team Aegle - University of Texas At Arlington - V 0.1

Figure 2-8 Mockup prototype for Item Request Form page

2.4 Definitions, Users, and Assumptions

Definitions:

- **Crate:** Defined as a plastic storage bin, which contains one or more items and can be stored in a shelf located in the storage room. Crates can be checked out from the storage room. Crates can have either one crate category or one project assigned to it. The crates have the following data fields in the system:
 - **Unique crate ID number.**
 - **List of items inside of the crate.**
 - **Crate location inside storage room.**
 - **Crate status:** A crate has two statuses.
 - **Inventory:** The crate is available inside the storage room.
 - **Out of Storage Room at User's Request:** The crate has been checked out of the storage room by a user. A request form must be submitted by the user and approved by an administrator before the crate can be set to this status. Request form defined later in this section.
 - **Crate Category or Project:** Optional field.
- **Item:** Defined as an object (i.e. Scissors) or collection of objects (i.e. 5 scissors in a bag, a box of Popsicle sticks, etc.) that may be stored inside a crate. Items can be checked out from the storage room. Every item must have an RFID tag. Items can optionally have either one crate category or one project assigned to it. Items can have up three alternate names, but only one main name. The items have the following data fields in the system:
 - **Unique item ID number.**
 - **Item name/s:** Some items may have multiple names.
 - **Item quantity.**
 - **Unique crate ID number:** Crate where the item is stored. Optional field.
 - **Item status:** An item has two statuses.
 - **Inventory:** The item is available inside the storage room.
 - **Out of Storage Room at User's Request:** The item has been checked out of the storage room by a user. A request form must be submitted by the user and approved by an administrator before the item can be set to this status. Request form defined later in this section.
 - **Project:** Optional field.
- **Request Form:** Defined as a form that is attached to an item or crate's system description when it goes from inventory to out of storage room at user's request. The form will contain the following data fields:
 - **Administrator:** Administrator who authorized the item or crate request.

- **Registered user:** Registered user who is submitting the request of the item or crate. Registered user could request items or crates on behalf of requesters.
 - **Requester:** A non-registered user on whose behalf a registered user is requesting an item for.
 - **Checkout Date:** The date on which an item or crate will be checked out of the storage room.
 - **Date of Return:** Planned date of item or crate return.
 - **Organization:** Name of the organization that the person who is requesting the item or crate belongs to.
 - **Reason:** The reason why the item or crate is being requested from the storage room.
- **Project:** A collection composed of one or more items required to perform a certain activity (i.e. “Flintstones”). Each project will maintain a list and quantity of the items required for the project.
 - **Crate category:** Crates can be organized with a crate category in order to store items of similar characteristics together (i.e. Crafts, Electronics, Tools, etc.).

Users:

- **Administrator:** The role of the administrator will be to manage the items and crates in the storage room. The administrator shall have the capability of adding and removing items, crates, and projects to/from the storage room. Administrators shall also be able to edit the data fields of items, crates, and projects stored in the system. In addition, the administrator also has the capability to deny or accept any request to check out items or crates to a registered user. The administrator will be able to view the location of the items and crates in the storage room. The system shall support multiple administrators, but only one main administrator (Dr.Tiernan) can add other administrators to the system.
- **Registered User:** The role of the registered user is to request or return items and crates to the storage room. They will also be able to view the current items or crates located in the storage room. The registered user shall only be able to retrieve items from the storage room after the administrator has accepted the item request.
- **Non-registered user:** A person wanting to register into the system.
- **Requesters:** Any person that requests an item or crate from the storage room. A requestor does not have to be a registered user.

- **Users:** Administrators, registered users, and non-registered users.

Assumptions:

- The registered user or administrator must place items being returned to the storage room in their respective crate if they belong to crate. If the item belongs to a crate with a crate category and the crate is empty, the item can be placed in a different crate. If there are remaining items of the same kind in the crate then the item must be placed in the crate that contains the remaining items.
- If an item is placed into a different crate it must be updated in the system.
- Each item shall have a unique RFID tag.
- Each crate shall have a unique RFID tag.
- The crate's material should not interfere with the radio frequency of the RFID reader.
- All items and crates must be initialized in the system.
- The client shall provide host computer.

3. Customer Requirements

The following section covers the requirements that the system shall provide to the user. OSS will be a smart inventory system capable of providing updates of the items in inventory with description of item, location, and quantity of the item.

3.1 Keep track Items and Crates by the System

3.1.1 Description: The system shall be able to keep track of the items' status (in stock or out of stock) and keep track of crates' status.

3.1.2 Source: Team Aegle.

3.1.3 Constraints: An item must have an RFID tags. The system must have a valid connection between the RFID reader and the server.

3.1.4 Standard: Federal Communication Commission (FCC) RFID bandwidth regulations and ISO-18000 radio frequency identification for item management.

3.1.5 Priority: 1 – Critical.

3.2 System Description of Items

3.2.1 Description: The system shall be able to provide a description of the item to the administrators and registered users. The description shall provide the item data fields stated previously on definitions section.

3.2.2 Source: Sponsor, Dr. Tiernan.

3.2.3 Constraints: Administrator or registered user must be logged in. A valid Internet connection to access the web application.

3.2.4 Standard: None.

3.2.5 Priority: 1 – Critical.

3.3 System Description of Crates

3.3.1 Description: The system shall be able to provide a description of the crate to the administrators and registered users. The description will provide crate data fields stated previously on definitions section.

3.3.2 Source: Sponsor, Dr. Tiernan.

3.3.3 Constraints: Administrator or registered user must be logged in. A valid Internet connection to access web application.

3.3.4 Standard: None.

3.3.5 Priority: 1 – Critical.

3.4 Search Function for Items, Crates and Projects.

3.4.1 Description: The system shall be able to search the database for items, crates and projects by the multiple data fields of the items and crates describe in the definitions section and by general word search.

3.4.2 Source: Sponsor, Dr. Tiernan.

3.4.3 Constraints: Administrator or registered user must be logged in. A valid Internet connection to access web application.

3.4.4 Standard: None.

3.4.5 Priority: 1 – Critical.

3.5 Locating Item Inside a Crate

3.5.1 Description: The system shall be able to locate in which crate an item is located.

3.5.2 Source: Sponsor, Dr. Tiernan.

3.5.3 Constraints: Item must have an RFID tag. A valid connection between the RFID reader and the server.

3.5.4 Standards: Federal Communication Commission (FCC) RFID bandwidth regulations and ISO-18000 radio frequency identification for item management.

3.5.5 Priority: 1 – Critical.

3.6 Item Management by the Administrators

3.6.1 Description: The administrators shall be able to add, remove and delete an item in the System.

3.6.2 Source: Sponsor, Dr. Tiernan.

3.6.3 Constraints: Valid Internet connection to access web application. Administrators must be logged in.

3.6.4 Standard: None.

3.6.5 Priority: 1 – Critical.

3.7 Crate Management by the Administrators

3.7.1 Description: The administrators shall be able to register a new crate into the system, as well as delete and edit crates already existing in the system.

3.7.2 Source: Team Aegle.

3.7.3 Constraints: Valid Internet connection to access web application. Administrators must be logged in.

3.7.4 Standard: None.

3.7.5 Priority: 2 – High.

3.8 Project Management by Administrators

3.8.1 Description: The administrators shall be able to create, edit and delete projects in the system.

3.8.2 Source: Sponsor, Dr. Tiernan.

3.8.3 Constraints: Valid Internet connection to access web application. Administrator must be logged in.

3.8.4 Standard: None.

3.8.5 Priority: 3 – Moderate.

3.9 System Interaction by Administrators

3.9.1 Description: The administrator shall be able to look at the items and crates in the inventory, look at the different projects, and shall be able to perform any functionality specified in other requirements. The administrator shall be able to approve or deny a registered user's request to check out an item(s) or crate(s) from the inventory. Also, the administrator shall be able to check out an item(s) or crate(s) from inventory and approve or deny a user registration request.

3.9.2 Source: Team Aegle.

3.9.3 Constraints: Administrators must be logged in. A valid Internet connection to access the web application.

3.9.4 Standard: None.

3.9.5 Priority: 1 - Critical.

3.10 System Interaction by Registered Users

3.10.1 Description: Registered users shall be able to look at the items in inventory, and shall be able to perform any functionality specified in other requirements. Registered users shall be able to look at the different projects, and submit a request for permission to take an item(s) or crate(s) from inventory, from the administrator.

3.10.2 Source: Sponsor, Dr. Tiernan.

3.10.3 Constraints: Registered user must be logged in. A valid Internet connection to access the web application.

3.10.4 Standard: None.

3.10.5 Priority: 2 – High.

3.11 Registration/Login System

3.11.1 Description: The system shall have a registration and login system, which will be the only way to access the database and system functionalities. The registration requirements will be an email account, first name, and last name, date of birth, phone number, organization, and password. The users shall be able to login into the system with their email account and password, after the administrator has approved their account.

3.11.2 Source: Team Aegle.

3.11.3 Constraints: A valid Internet connection to access the web application.

3.11.4 Standard: None.

3.11.5 Priority: 1 – Critical.

3.12 Web-Based Accessible Application

3.12.1 Description: The system shall be implemented as a web-based application.

3.12.2 Source: Team Aegle.

3.12.3 Constraints: Google Chrome and Mozilla Firefox shall be the only browsers supported by the website application.

3.12.4 Standard: IETF Standards.

3.12.5 Priority: 1 – Critical.

3.13 Locating Crate Inside the Storage Room

3.13.1 Description: The system shall be able to provide a relative location for a crate inside the storage room.

3.13.2 Source: Sponsor, Dr. Tiernan.

3.13.3 Constraints: Crate must have an RFID tag. A valid connection between the RFID reader and the server.

3.13.4 Standard: None.

3.13.5 Priority: 4 – Low.

4. Packaging Requirements

This section specifies all of the packaging requirements that must be met by OSS. A user manual with all the software required will be provided in order to do self-installation of the RFID system.

4.1 Included Hardware Components

4.1.1 Description: The final product shall include the following components: Passive RFID tags for the items and an RFID reader.

4.1.2 Source: Team Aegle.

4.1.3 Constraints: None.

4.1.4 Standards: None.

4.1.5 Priority: 1 – Critical.

4.2 Installation Manual

4.2.1 Description: The final product shall provide an installation manual that includes detailed instructions on how to install, set up, and use the system.

4.2.2 Source: Team Aegle.

4.2.3 Constraints: None.

4.2.4 Standards: None.

4.2.5 Priority: 3 – Moderate.

4.3 Range of the RFID Reader Integrated Antenna

4.3.1 Description: The RFID reader antenna range shall be able to cover the door height.

4.3.2 Source: Team Aegle.

4.3.3 Constraints: The RFID reader with the integrated antenna must be placed in a static position within the doorframe.

4.3.4 Standards: None.

4.3.5 Priority: 2 – High.

4.4 Software Components

4.4.1 Description: The final product shall include the following software components: source code. Shall be delivered on a USB flash drive.

4.4.2 Source: Team Aegle.

4.4.3 Constraints: None.

4.4.4 Standards: None.

4.4.5 Priority: 1 – Critical.

4.5 RFID Reader Installation

4.5.1 Description: The RFID reader shall be mounted above the door enclosed in a frame. A technician shall not be required to install the reader.

4.5.2 Source: Team Aegle.

4.5.3 Constraints: None.

4.5.4 Standards: None.

4.5.5 Priority: 2 – High.

4.6 Software Installation

4.6.1 Description: The software shall be delivered in an installation script guiding the user through the necessary steps to complete the installation.

4.6.2 Source: Team Aegle.

4.6.3 Constraints: None.

4.6.4 Standards: None.

4.6.5 Priority: 4 – Low.

5. Performance Requirements

The following section covers the performance requirements for the project. The primary concerns for performance in our project lies in the responsiveness of the interface and the usability of the product in conditions where the system is deprived of resource that it uses to operate. It is expected that the product will have a responsive and streamlined feel to it, which these requirements mandate.

5.1 Check-in/Check-out Latency

5.1.1 Description: The amount of time that it takes for the system to recognize that an item has entered or left the storage room should not exceed 1 minute.

5.1.2 Source: Team Aegle.

5.1.3 Constraints: Network connectivity and speed.

5.1.4 Standards: None.

5.1.5 Priority: 3 – Moderate.

5.2 Web Interface Response Latency

5.2.1 Description: The amount of time that it takes for the web application to return meaningful information to the user shall not exceed 10 seconds.

5.2.2 Source: Team Aegle.

5.2.3 Constraints: Network connectivity and speed.

5.2.4 Standards: None.

5.2.5 Priority: 3 – Moderate.

6. Safety Requirements

The following section outlines the requirements that protect the system's end users from physical and monetary damages related to OSS.

6.1 Electrical Hazard

6.1.1 Description: The system and its components present within the storage room shall not pose an electrical hazard to its users or the building it resides in.

6.1.2 Source: Team Aegle.

6.1.3 Constraints: None.

6.1.4 Standards: None.

6.1.5 Priority: 1 – Critical.

6.2 Signal Interference

6.2.1 Description: The RFID system shall not interfere with any critical radio frequency transmission.

6.2.2 Source: Team Aegle.

6.2.3 Constraints: FCC signal interference guidelines.

6.2.4 Standards: None.

6.2.5 Priority: 3 – Moderate.

7. Maintenance and Support Requirements

The following section specifies the features and help that the end user will be provided with upon product completion. It is not expected that the developers of this product will provide any personal support after the project's completion, but it is expected that the end users will have the necessary documentation and instructions to fix a simple problem, or when a third party needs to be involved.

7.1 User Manual

7.1.1 Description: The final product shall come with a user manual describing in detail how to set up the system and use its various features.

7.1.2 Source: Team Aegle.

7.1.3 Constraints: None.

7.1.4 Standards: None.

7.1.5 Priority: 3 – Moderate.

7.2 Troubleshooting Guide

7.2.1 Description: The final product shall include a troubleshooting guide to help solve general problems the user may have, and to assist them in determining whether the problem needs to be solved by a third party.

7.2.2 Source: Team Aegle.

7.2.3 Constraints: None.

7.2.4 Standards: None.

7.2.5 Priority: 4 – Low.

7.3 Source Code Availability & Documentation

7.3.1 Description: The final product shall include all the source code and documentation used to design and implement the system. The source code will be well structured and commented so as to allow for future modularity and support.

7.3.2 Source: Team Aegle.

7.3.3 Constraints: None.

7.3.4 Standards: None.

7.3.5 Priority: 3 – Moderate.

7.4 Database Interchangeability

7.4.1 Description: The final product shall be designed in such a way that different databases (such as SQL, Oracle, etc.) can be adapted without the need of any major changes to the source code. However, the team will only be responsible for developing an SQL database.

7.4.2 Source: Team Aegle.

7.4.3 Constraints: None.

7.4.4 Standards: None.

7.4.5 Priority: 4 – Low.

8. Other Requirements

The following section describes all the requirements that did not fall directly under any of the previous sections of this document will be found here.

8.1 Information Security

8.1.1 Description: The system shall ensure the privacy and security of personal information being stored and transmitted by users of the system through the use of secure connectivity and secure programming techniques.

8.1.2 Source: Team Aegle.

8.1.3 Constraints: None.

8.1.4 Standards: None.

8.1.5 Priority: 3 – Moderate.

8.2 Connectivity Tolerance

8.2.1 Description: The system shall be tolerant of a loss of communication with the web server. During loss of connectivity to the server, transactions shall be stored in the buffer of a user's device. Transactions shall be processed when connectivity to the server is reestablished.

8.2.2 Source: Team Aegle.

8.2.3 Constraints: End user device's storage buffer maximum size is 16 KB of data.

8.2.4 Standards: None.

8.2.5 Priority: 2 – High.

9. Acceptance Requirements

The following section describes the different procedures that validate the correct functionality of OSS to the customer. Every acceptance requirement should be valid in order for the product to be approved by Aegle and our customer. As the project develops, the criteria will be redefined to accommodate any upcoming modifications.

9.1 Verify User Account Registration

9.1.1 Requirements Addressed: Requirements 3.9, 3.10, 3.11, and 3.12. System must be able to verify that a new user can be registered and be able to login.

9.1.2 Verification Procedure: The Home page in the web application will have a “Register” button that allows the user to send a registration request to the administrator, who will either accept or deny the user to be added to the system. After the administrator logs into the system and he will navigate to the “User Management” section, where a list of user requests will be displayed. When a user is selected, the administrator will click on the “Accept” button and the user should be listed in the approved users list. Lastly, the new user will successfully log in to the system using email and password.

9.2 Verify Location of an Item Functionality

9.2.1 Requirements Addressed: Requirements 3.1, 3.2, 3.4, 3.5, 3.9, 3.10, 3.11, and 3.12. System must be able track an item in the storeroom and locate which crate it is stored in.

9.2.2 Verification Procedure: This requirement will be verified by searching on a registered item that will accurately display the crate where the item is contained. First, an authorized user will login to the system and click on the “Search” menu in the “Dashboard Management” page. The user will be redirected to the “Search” page, where search criteria will be specified. After clicking on the “Search” button, the search results will be displayed with a list of items that matched the search criteria. Clicking on the button “See details” will show a more detailed description of the selected item.

9.3 Verify Location of a Crate Functionality

9.3.1 Requirements Addressed: Requirements 3.1, 3.3, 3.4, 3.9, 3.10, 3.11, 3.12, and 3.13. System must be able to track a crate inside the storeroom, including its contents and its location.

9.3.2 Verification Procedure: This requirement will be verified by searching on a registered crate that will display the shelf where the crate is placed. First, an authorized user will login to the system and click on the “Search” link in the navigation bar. The user will be redirected to the “Search” page, where search criteria will be specified. After clicking on the “Search” button, the crate will display information of where the crate is located, providing a description and the shelf where the crate is located.

9.4 Verify Search Functionality

9.4.1 Requirements Addressed: Requirements 3.1, 3.4, 3.5, 3.9, 3.10, 3.11, 3.12, and 3.13. System must be able to search for an item in the storeroom, including which crate it is located in if it has one.

9.4.2 Verification Procedure: The “Search” button in the web application will search for an item in the storage room. First, the administrator or registered user will log in to their respective accounts. Next, the user will click on the “Search” link, located in the navigation bar. The user will then be redirected to the “Search” page where different search criteria will be selected and correctly filled out. Lastly, the user will click on the “Search” button and the system will display the items, crates or projects along with the corresponding location.

9.5 Verify Item Management Functionality

9.5.1 Requirements Addressed: Requirements 3.2, 3.4, 3.5, 3.6, 3.9, 3.11 and 3.12.

System must be able to manage items stored in the storeroom such as adding and removing items.

9.5.2 Verification Procedure: An authorized user must register an item into the system along with the corresponding crate that will contain it if it has one. Once the item is successfully registered into the system, a search will be performed to make sure the item is correctly displayed by the system. Next, the authorized user will select the item in the search results and click on the “Edit” button. The user will be redirected to the “Edit Item” section, where they will be able to edit the item name or description. After editing, the user clicks on the “Save” button. The item description or name should display the new information. Lastly, the user will search for the item again. On the search results, the item will be selected and the user will click on the “Remove” button. A confirmation alert shall be displayed with two buttons: a “Cancel” button that will cancel the deletion of the item and a “Remove” button that will permanently remove the item from the system. If a new search is performed for the item, the item should no longer be displayed in the system.

9.6 Verify Crate Management Functionality

9.6.1 Requirements Addressed: Requirements 3.3, 3.4, 3.7, 3.9, 3.11, 3.12, and 3.13.

System must be able to manage crates in the storeroom including adding and removing crates.

9.6.2 Verification Procedure: An authorized user must register a crate into the system along with the corresponding shelf where the crate will reside. Once the crate is successfully registered in the system, a search will be performed to make sure the crate is successfully registered in the system. Next, the authorized user will select the crate in the search results and click on the “Edit” button. The user will be redirected to the “Edit Crate” section, where he will be able to edit the crate name or description. After editing, the user clicks on the “Save” button. The crate description or name should display the new information. Lastly, the user will search for the crate again. On the search results, the crate will be selected and click on the “Remove” button. A confirmation alert will be displayed with two buttons: a “Cancel” button that will cancel the deletion of the crate and a “Remove” button that will permanently remove the crate from the system. If a new search is performed in the crate, the crate should no longer be displayed in the system.

9.7 Verify Project Management Functionality

9.7.1 Requirements Addressed: Requirements 3.4, 3.8, 3.9, 3.11 and 3.12. System must be able to manage projects such as adding, removing and editing projects.

9.7.2 Verification Procedure: An authorized user must login to the system and navigate to the “Projects” page. Once the user has navigated to the page, they will click on the “Add New Project” button and fill out the project form. Once the form is correctly filled out, the user will click on the “Save” button. The project will display in the projects list. The user will select the recently created project and click on the “Edit Project” button. The system will display the project form where the user will edit information and click on the “Save” button. The project changes will be displayed correctly. Finally, the user will select the same project and click on the “Remove” button. A confirmation alert will be displayed with two buttons: a “Cancel” button that will cancel the deletion of the crate and a “Remove” button that will permanently remove the project from the system. The project will no longer be visible in the projects list.

10. Use Cases

The following section will cover how the users interact with OSS via the web-based application. The use cases will assume that the system has been installed, setup and ready for use.

Definitions: TUCBW: This Use Case Begins With. TUCEW: This Use Case Ends With.

10.1 Registering into the System

10.1.1 Scenario: A non-registered user clicks on the “Log In / Register” button while in the web-based application. This action will redirect the person to the log in / registration webpage. In this webpage, the non-registered user must fill in the information and click complete registration in order to complete the registration process. The following information is needed by the non-registered user in order to register: email, first name, last name, date of birth, phone number, organization, and password.

10.1.2 Actor(s): Non-registered users.

10.1.3 TUCBW: The non-registered user clicks on the register button.

10.1.4 TUCEW: The non-registered user is notified that the registration was submitted.

10.2 Logging into the System

10.2.1 Scenario: Either an administrator or registered user is on the home page of the web-based application. They fill in the login box information with their email and password, and click on Log In or press Enter to complete the login process.

10.2.2 Actor(s): Administrators or registered users.

10.2.3 TUCBW: The administrator or registered user opens the web-based application.

10.2.4 TUCEW: The administrator or registered user is logged in to the system.

10.3 Adding Item, Crate, or Project to the System

10.3.1 Scenario: The administrator clicks on the “Add Item” button, “Add Crate” button, or “Add Project” button while on the inventory webpage. This will prompt the administrator to fill out a form for the new item, crate, or project to be added into the system. The administrator must fill out the form and click complete. After clicking complete the administrator will be redirected to the list of items, list of crates, or list of projects with the newly added item/crate/project highlighted and showing the item/crate/project description below it.

10.3.2 Actor(s): Administrators

10.3.3 TUCBW: The administrator clicks on “Add Item” button, “Add Crate” button, or “Add Project” button.

10.3.4 TUCEW: The administrator sees the item, crate, or project and its description in the item list, crate list, or project list.

10.4 Editing Item, Crate, or Project in the System

10.4.1 Scenario: The administrator clicks on the “Edit” button either for an item in the items list, for a crate in the crates list, or project in the projects list. After this action the administrator is redirected to a website where it can make any changes to the item/crate/project, exceptions are to change the ID# of an item or crate. After making the changes, the administrator must click on complete in order to finish editing. After that the administrator is redirected to the list of items, list of crates or list of project.

10.4.2 Actor(s): Administrators.

10.4.3 TUCBW: The administrator clicks on the “Edit” button for an item, crate, or project.

10.4.4 TUCEW: The administrator is redirected to the list of items, crates, or projects.

10.5 Deleting Item, Crate, or Project from the System

10.5.1 Scenario: The administrator clicks on the “Delete” button either for an item in the items list, for a crate in the crates list, or project in the projects list. After this action the administrator will be asked for a final confirmation. After clicking either Yes or No, the administrator is redirected to the items list, crates list, or projects list.

10.5.2 Actor(s): Administrators.

10.5.3 TUCBW: The administrator clicks on the “Delete” button for an item, crate, or project.

10.5.4 TUCEW: The administrator is redirected to the list of items, crates, or projects webpage.

10.6 Searching for Items and Crates

10.6.1 Scenario: The administrator or registered user types in a word in the search box and presses Enter or clicks on the “Search” button. Another way to search is by clicking advanced search, filling in the search criteria, and clicking on the “Search” button. Once the administrator or registered user clicks on the “Search” button, they are directed to the search results.

10.6.2 Actor(s): The administrator or registered users.

10.6.3 TUCBW: The administrator or registered user does a search by word or an advanced search.

10.6.4 TUCEW: The administrator or registered users are shown the search results.

10.7 User Registration Confirmation by Administrator

10.7.1 Scenario: The administrator will receive email notifications about user registration that they will have to approve or deny. The administrator can review the registration before denying or approving it. The administrator will click on the pending registrations link and be taken to a webpage where they can view the user registration information and then click on the “Approve” or “Deny” button.

10.7.2 Actor(s): Administrator.

10.7.3 TUCBW: Administrator clicks on the pending registrations link.

10.7.4 TUCEW: Administrator is notified that the registration was successfully approved or denied.

10.8 Item/Crate Check-out Approval by Administrator

10.8.1 Scenario: The administrator will receive email notifications about registered users who request to checkout item(s) or crate(s) from the storage room. The administrator will have to approve or deny the request. The administrator will click on the pending requests link to see a list of requests. After clicking on a specific request the administrator will be shown the request form and must click on the “Approve” or “Deny”.

10.8.2 Actor(s): Administrator.

10.8.3 TUCBW: The administrator clicks on the pending request link.

10.8.4 TUCEW: The administrator is notified that the request has been successfully approved or denied.

10.9 Item/Crate Checkout Request by Registered User

10.9.1 Scenario: The registered user can submit item(s) or crate(s) checkout requests to the administrator by selecting the item(s) or crate(s) and then clicking on the “Checkout” button. The registered users must fill out the request form.

10.9.2 Actor(s): Registered users.

10.9.3 TUCBW: The registered user click on the “Checkout” button after selecting the item(s) or crate(s) to be checked out.

10.9.4 TUCEW: The request is sent to the administrator.

10.10 Item/Crate Checkout by Administrator

10.10.1 Scenario: The administrator selects the item(s) or crate(s) that they want to check out and then they click on the “Checkout” button. The administrator must fill out a request form after clicking on the “Checkout” button, and then click Confirm to finish the checkout.

10.10.2 Actor(s): Administrator.

10.10.3 TUCBW: The administrator selects the item(s) or crate(s) and clicks on the “Checkout” button.

10.10.4 TUCEW: Administrator sees a confirmation message.

10.11 Item/Crate Check-in by Administrators and Registered Users

10.11.1 Scenario: The administrator or registered user returning an item must place the item in its respective crate if it belongs to one, and update the quantity of the item if necessary in the web application. Also, if the administrator or registered user places the item in a different crate, they must update the item location change in the web application. Returning a crate will be just as returning an item, the quantity of the items inside the crate must be updated if necessary and if the location of the crate changed then the crate location must also be updated in the web application.

10.11.2 Actor(s): Administrators and registered users.

10.11.3 TUCBW: The administrator or registered user brings an item or crate into the storage room.

10.11.4 TUCEW: The administrator or registered user gets a notification that the item or crate was successfully returned.

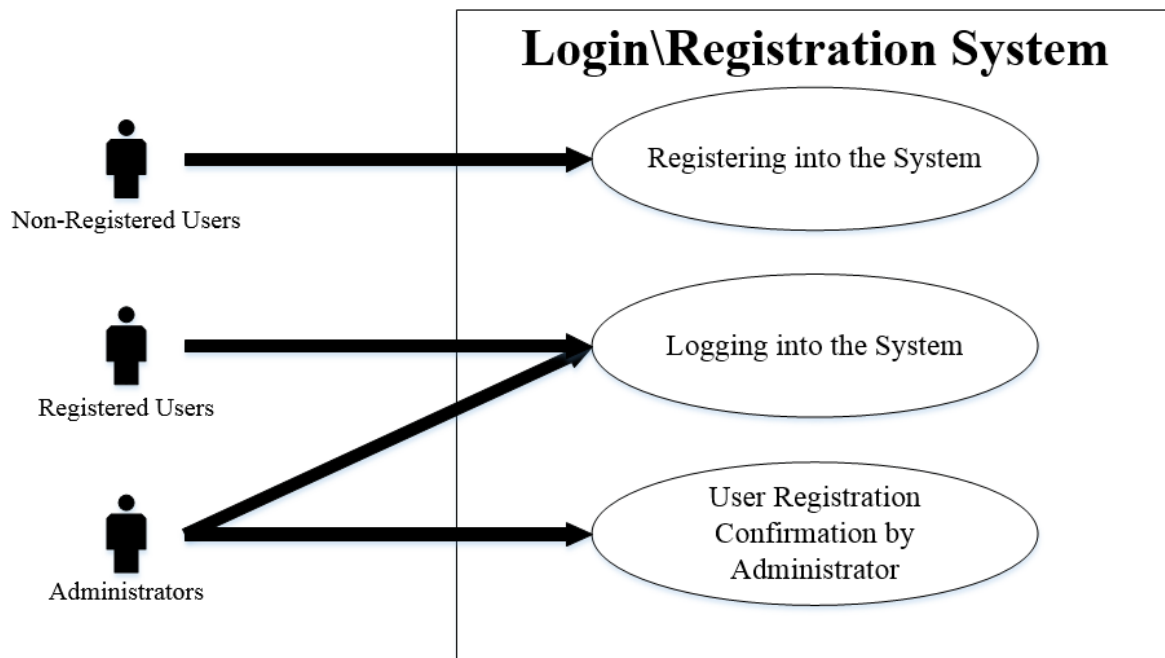


Figure 10-1 Login and Registration System

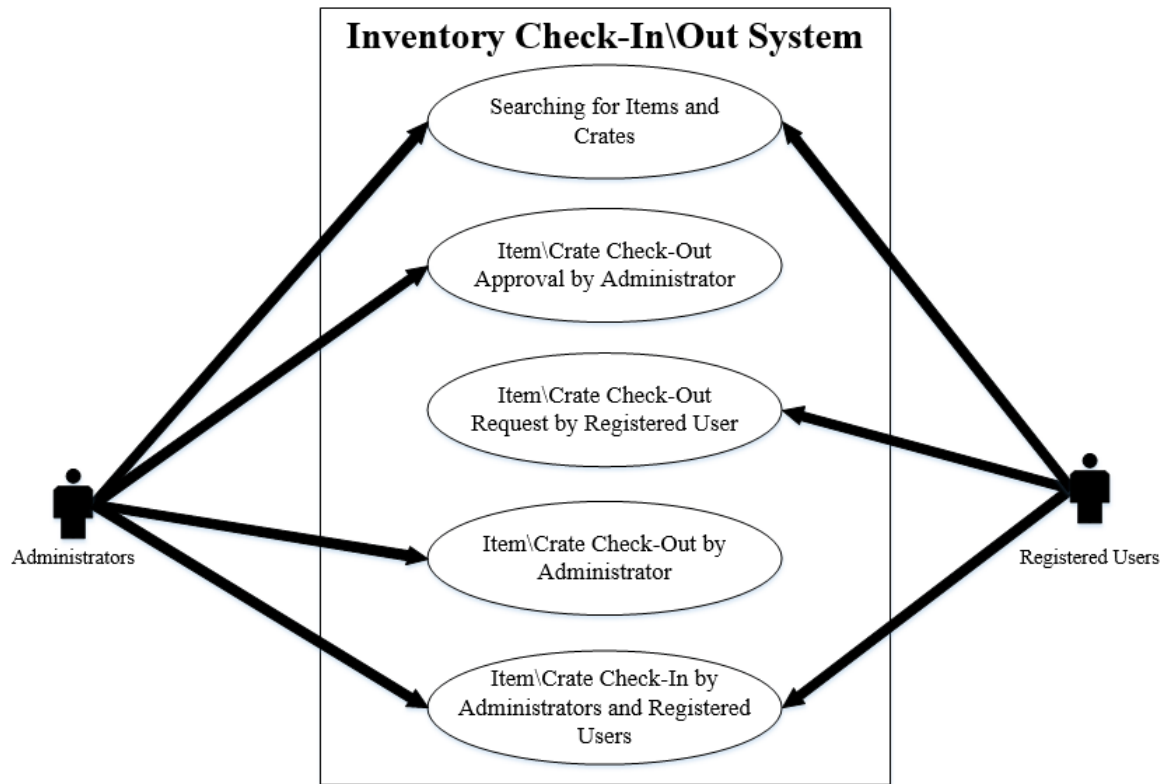


Figure 10-2 Inventory Check In / Out system

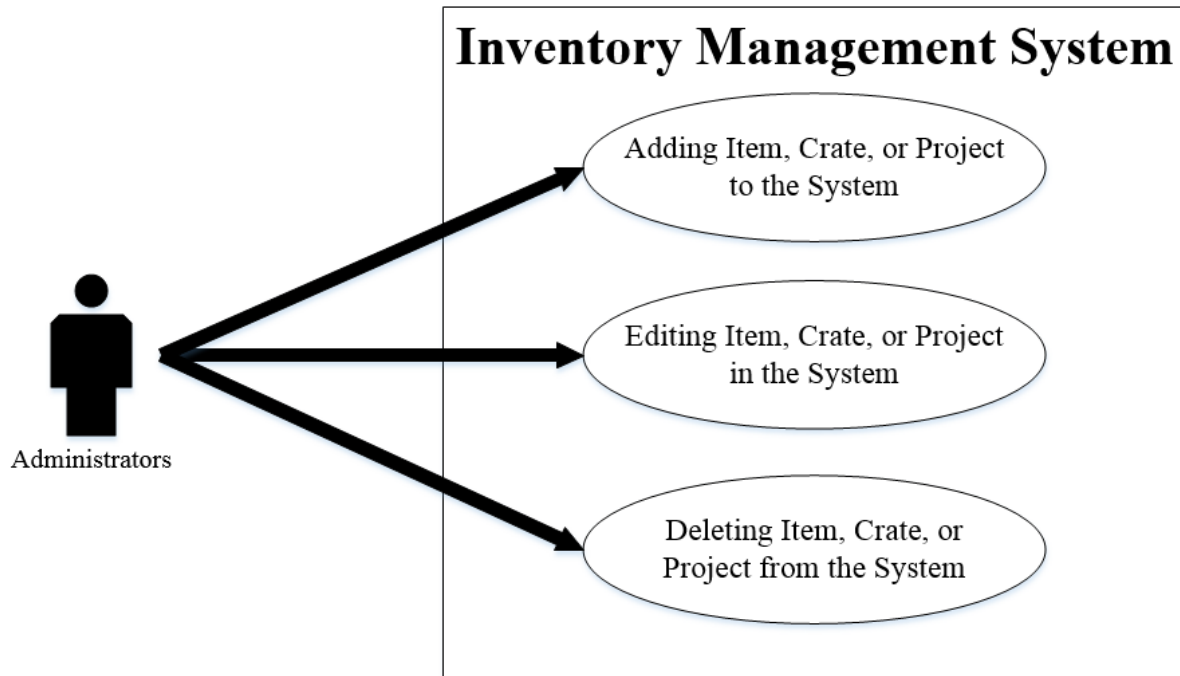


Figure 10-3 Inventory Management

11. Feasibility Assessment

The following section provides a Feasibility Assessment of OSS, according to Aegle's estimation experience acquired in previous projects. The assessment is based on six different sections: Scope Analysis, Research, Technical Analysis, Cost Analysis, Resource Analysis and Schedule Analysis.

11.1 Scope Analysis

The scope of OSS is based on the critical requirements provided by the sponsor and is reasonable given the time frame, budget and resources available. The prototype system will be composed of an RFID reader with an integrated antenna, RFID tags, crates, items, and a server (that will be provided by the user). The software will be installed on the server (Team Aegle will use their personal server for testing). The most critical requirement is being able to maintain inventory count of the items in the storage room by successfully sending data from the RFID reader through a network and registering correctly in the server. The web application has to be able to display all of the items or crates in the storeroom and their location. Requirements are ordered by priority, from most critical to low priority, to ensure that critical requirements are addressed in the early stages of development in case any issue arises.

11.2 Research

Our development team has done intensive research into RFID technology in order to understand how data is passed through radio frequency waves and how we can transmit this data through a network. Defining the type of RFID reader that we will have to implement is very important to our project, since it has a high cost associated with it; therefore an informed decision must be made. We have identified different types of RFID readers that can be implemented in the storage room; they can be a stationary or a handheld device. Further research on this technology, is indicating that a stationary RFID reader will be the most appropriate reader for our needs. This will facilitate the readability of multiple RFID tags at the same time without reader collision problems.

The University of Texas at Arlington has an RFID research laboratory managed by Dr. Jones and Ph.D. students who contribute to the research activities. The team has met numerous times with various Ph.D. students in the lab in order to gain a better understanding of this technology to help create a better design of the hardware portion of this system.

As a result of the research performed by the team, OSS will need a stationary RFID reader with an integrated antenna, RFID passive tags, a valid network connection, and a host server where the software will be installed. These pieces will be critical for the development of the project.

11.3 Technical Analysis

The type of RFID reader that the team will implement will dictate the manner in which the items or crates will be registered into the application. Therefore it is one of the most important decisions that the team has to make since most of the critical requirements depend on the RFID reader being able to read and send data through a network transferred in packages. According to the research that the team has done, the team believes that a stationary RFID reader is the most viable for the type of operations that must be performed in the storage room.

The OSS application shall be responsible for receiving and parsing any data that the RFID reader transmits through the network in standard format such as JSON, Binary Data, XML, etc.

The items, crates and shelves will be assigned a unique RFID passive tag. Passive tags have an infinite lifespan due to the fact that the RFID reader powers the tag's internal battery when the reader is in close range.

The system will integrate an open source API named "Mercury"; which will be responsible for receiving and processing the data obtained by the RFID reader. The OSS system will also have an internal Restful API, which will communicate with Mercury to receive the tag's data and do the necessary processing. This Restful API will also communicate with the web client application, providing the customer with a visible state of the inventory. The server that the team will be using for testing is going to run on Linux.

The web client application will also allow different types of users to perform a variety of tasks, depending on the user's privileges. The web client application will be able to modify the state of the database if it is operated through an authorized system.

11.4 Cost Analysis

According to the cost research that our team has done, it has been determined that the project can be successfully developed within the eight hundred dollars budget that the team has been allocated. The project is composed of both hardware and software. The hardware part of the project includes the RFID reader with an integrated antenna, RFID passive tags, items and crates that will be used to test the project functionality. The software part of the project needs a server to host the restful API and the web client application.

The following table shows relative prices of the components required for the development of OSS.

Parts	Cost
RFID Reader	\$445.00
RFID Tags (30)	\$20.00
Miscellaneous items	\$30.00
Plastic crates (4)	\$40.00
Total Cost	\$535.00

Table 11-1 Preliminary Cost Analysis

11.5 Resource Analysis

Team Aegle consists of two software engineering students, one computer-engineering student and one computer science student.

The OSS is composed of two main branches, hardware and software. The hardware branch of the project consists of reading RFID tags, and correctly configuring the RFID reader so that it can correctly transmit data through a network. Our computer-engineering student, who has more experience in hardware technology, will lead this work. The computer science student has been exposed to some hardware; therefore he will be focused in both hardware and software since this part corresponds to around a 30% of the entire application.

As far as the software project, the two software engineering students will be mainly focused on developing the entire application, dividing the work in two, which will be the web client interface and the server side application. Finally, the computer science student will be the database implementation lead, due to his experience with relational databases in previous projects.

The fact that each team member has a role and a part of the application assigned doesn't mean they are restricted to only work on that part of the application. When it comes to implementing new technologies and doing research, the whole team will be involved to make sure we all share the same knowledge and we are all in the same terms.

All our team members have experience writing software and doing architectural and database design, we feel very strong and comfortable on this part of the project, since architectural design and coding was one of our strengths and our project is heavily based in software.

As far as our weaknesses, only one of the team members has knowledge working with hardware components, so having only one team member experienced in this could potentially delay the development schedule. However, we will maintain a risk management plan and perform risk monitoring to ensure we stay on track and we are able to identify any potential issues related to hardware as soon as possible before they cause catastrophic damage to our project.

11.6 Schedule Analysis

The following section will talk about three different techniques that the team used to estimate how complex the project is and the time it would take to develop it, according to different factors that influence the design and implementation of OSS.

The first method of estimation that we implemented is the Jones First Order. The purpose of this method is to identify how big or complex the project is, in order for us to determine if we are able to successfully complete the project on the time frame available.

First, we identified all the inputs, outputs, inquiries, logical internal files and external interfaces. After that, we classify them as low complexity, medium complexity or high complexity, as demonstrated on the table below.

Function Type	Low Complexity	Medium Complexity	High Complexity
Number of inputs	12 x 3	9 x 4	0 x 6
Number of outputs	4 x 4	3 x 5	0 x 7
Inquiries	0 x 3	1 x 4	0 x 6
Logical internal files	5 x 7	3 x 10	0 x 15
External interface files	0 x 5	0 x 7	1 x 10
Unadjusted function-point total			182

Table 11-2 Function Point Table Breakdown

The previous table demonstrates the number of different functions that team Aegle identified as being part of the Outreach Storage System, with the intention of producing the unadjusted function point total, which turned out to be one hundred and eighty two.

After calculating the unadjusted functional point total, we assigned values to the influence multipliers, with the purpose of getting a more accurate estimation.

Adjustment factor	Degree of Influence (0-5)
Data Communication	5
Distributed Data Processing	0
Performance	1
Heavily Used Configuration	0
Transaction Rate	2
On-line Data Entry	4
End-User Efficiency	3
On-line Update	5
Complex Processing	1
Reusability	0
Installation Ease	3
Operational Ease	2
Multiple Sites	0
Facilitate Change	2
Sum	27

Table 11-3 Influence Multipliers

As demonstrated by the table 11-3, the total value of the Influence Multiplier tuned out to be of twenty-seven.

After getting an adjustment factor, we calculated the influence multiplier with the following formula:

$$\text{Influence Multiplier} = (27 * .01) + .65 = 0.92$$

Multiplying the Influence Multiplier by our Unadjusted Function Point total, we obtained our Adjusted Function Point total.

$$\text{Adjusted Function Point Total} = 0.92 * 182 = 167.44$$

Lastly, we use the Jones First Order estimation procedure to calculate the time it would take to develop a project using the established influence multiplies and unadjusted function points. OSS falls under the “Systems” category, which is analyzed in Best Case, Average Case and Worst Case scenarios, the calculations are displayed in the following table.

$$\text{Duration} = 172.9^{0.45} = 10.01 \text{ Calendar Months}$$

	Best Case	Average Case	Worst Case
Adjusted Function Point	167.44 ³⁹	167.44 ⁴²	167.44 ⁴⁵
Total	7.36	8.59	10.01

Table 11-4 Jones First Order Estimation

The data in Table 11-4 shows that at our very best the project will take approximately seven and a half months to complete and in the worst case scenario it could take up to 10 months.

The second estimation model is the Simplified Hybrid Approach. We decided to take this approach because using function points for estimation is typically more accurate. The following table calculates the estimated the lines of code of according to our function points based on the QSM Function Points Languages Table.

Function Type	Language	Approximate LOC/ Function Point	Total
Number of inputs	HTML	34 x 72	2,448 LOC
Number of outputs	HTML	34 x 21	714 LOC
Inquiries	SQL	21 x 4	84 LOC
Logical internal files	Java	65 x 53	3, 445 LOC
External interface files	Java	65 x 10	650 LOC
Total	7,141 LOC		

Table 11-5 Simplified Hybrid Approach

The data in Table 11-4 describes how the SLOC calculations for the different function points were calculated, based on QSM Function Points Language Table. Producing a result which indicates that at our very best the project will take approximately seven and a half months to complete and worst case scenario it could take up to 10 months.

After calculating the estimated LOC we now need to calculate the effort, which is calculated based on the COCOMO simplified formula, which multiplies the KSLOC by a factor of one point four:

$$\text{Effort} = 1.4 * 7.141 = 9.99$$

Finally, we made use of the Rule of Thumb formula to calculate the duration of the project:

$$\text{Time} = 3.0 * 9.99^{1/3} = 6.46 \text{ months}$$

The third estimation model that we decided to use was the Sanity Test method by Weiss and Wysocki. For this sanity test, we used the results gathered from the Simplified Hybrid Approach, hoping to get similar results so that we can get a more accurate estimation. To calculate sanity test, we need to make use of an optimistic duration which in this case is the result that we obtained by the Hybrid Approach (6.46 months). For the nominal duration, we used the best case obtained in Jones' First Order Model (7.36 months). The third estimate we used is the pessimistic duration, also obtained by the Jones' First Order method (10.01 months).

The following formula represents the Sanity Test calculations and results.

$$E = O + 4M + P / 6$$

$$E = (6.46 + 4(7.36) + 10.01) / 6 = 7.6 \text{ Months}$$

The sanity test results indicate that according to our calculations, we would need seven and a half months to finish OSS. Our team will be able to implement the most critical and high importance requirements within this period. Low priority items will be developed towards the end of the cycle, to ensure that critical items are fully developed in the given time frame.

12. Future Items

All of the requirements listed below will be implemented as time and schedule permit.

12.1 Automatic Item Location Per Crate Inside the Storage Room

12.1.1 Description: The system shall automatically track items contained within each individual crate.

12.1.2 Constraints: Cost and time. The budget that is allocated to the team is not sufficient to account for costs related to individual crate tracking, such as sensors or other technology types. Also, the time budgeted is not be enough for the team to implement this functionality, therefore the team has agreed on implementing a fast mechanism to track the items, trying to increase the performance of the system.

12.2 Automatic Crate Location Inside the Storage Room

12.2.1 Description: The system shall be able to provide an exact location of a crate inside the storage room without user interaction.

12.2.2 Constraints: Cost. With the current budget, it is impossible to buy three extra antennas and a more expensive reader to support them.

12.3 Mobile Application

12.3.1 Description: The user shall be able to manage OSS using a mobile application.

12.3.2 Constraints: Time. The time that the team has for the completion of the project is only enough to develop a web application. If the team happened to develop a mobile application, the project plan will be in jeopardy causing some tasks to be incomplete. The team has decided to develop a responsive web application, so that it can be launched in a mobile browser.